80

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/850,183	05/07/2001	Mark A. Kampe	80168-0106	9753

32658 7590 06/27/2005

HOGAN & HARTSON LLP
ONE TABOR CENTER, SUITE 1500
1200 SEVENTEEN ST.
DENVER, CO 80202

EXAMINER
PROCTOR, JASON SCOTT

ART UNIT	PAPER NUMBER
2123	

DATE MAILED: 06/27/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No. 09/850,183	Applicant(s) KAMPE, MARK A.	
	Examiner Jason Proctor	Art Unit 2123	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.
- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *14 April 2005*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-16,18 and 19* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-16,18 and 19* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *23 February 2005* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)

2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.

4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.

5) ☐ Notice of Informal Patent Application (PTO-152)

6) ☐ Other: _____.

DETAILED ACTION

Request for Continued Examination

1. A request for continued examination under 37 CFR 1.114, including the fee set forth in 37 CFR 1.17(e), was filed in this application after final rejection. Since this application is eligible for continued examination under 37 CFR 1.114, and the fee set forth in 37 CFR 1.17(e) has been timely paid, the finality of the previous Office action has been withdrawn pursuant to 37 CFR 1.114. Applicant's submission filed on February 23, 2005 has been entered.

Claims 1, 8-12, 16, and 18-19 have been amended and claim 17 has been cancelled in Applicants' response dated February 23, 2005. Claims 1-16 and 18-19 are pending in the application.

Response to Objections to the Drawings

The Examiner thanks Application for the submission of a replacement drawing sheet in response to the previous objections to the drawings. The previous objections have been withdrawn.

Response to Rejections under 35 U.S.C. § 112

Regarding the rejection of claims 8-15, 17, and 18 under 35 U.S.C. § 112, second paragraph, as being indefinite, the Examiner thanks Applicant for providing concise definitions of the terms "warm recoverable errors" and "non-warm recoverable errors" in the claim language. The Examiner thanks Applicant for clarifying or amending the claim language to resolve these

rejections. The previous rejections under 35 U.S.C. § 112, second paragraph, have been withdrawn.

Response to Rejections under 35 U.S.C. § 102

Regarding the rejections of claims 1-5, 16, and 19 under 35 U.S.C. § 102(e) as being anticipated by US Patent No. 6,393,386 to Zager et al. (Zager), Applicants argue primarily that:

> However, none of this discussion nor any citations to Zager discuss the specific software failure or error classes now claimed as being modeled by aggregated failure rates in claim 1. [...] However, Zager only teaches root and non-root fault classifications and such a classification is not inherently useful with software application errors and is different than that claimed in claim 1.

The Examiner has fully considered these arguments, and in light of the amendments to the claim language, finds them persuasive. The previous rejections under 35 U.S.C. § 102(e) based on the Zager reference have been withdrawn.

Response to Rejections under 35 U.S.C. § 103

Regarding the rejections of claims 6-15, 17, and 18 under 35 U.S.C. § 103(a) as being unpatentable over Zager in view of "Understanding Fault-Tolerant Distributed Systems" by Flaviu Cristian (Cristian), Applicants refer to the arguments in favor of claim 1 as applicable, and further that Cristian is not cited to overcome the deficiencies of the Zager reference.

The Examiner has fully considered these arguments, and in light of the amendments to the claim language, finds them persuasive. The previous rejections under 35 U.S.C. § 103(a) based on Zager in view of Cristian have been withdrawn.

Terms Defined in the Art

In the interest of expediting discussion of the prior art, the Examiner refers to "Survey of Software Tools for Evaluating Reliability, Availability, and Serviceability", by Allen M. Johnson, Jr., and Miroslaw Malek (Malek) to define the relationship between "availability graphs" and "reliability graphs".

> Availability graphs are recognizable by the condition that no system-fail state (sometimes called a trapping or death state) is an absorbing state (i.e., there is a repair rate linking back to some previous operating state). Conversely, reliability graphs are conspicuous by the fact that all system-fail states are absorbing states with no links directed out. (page 241, right column – page 242, left column).

The Examiner remarks that the difference between availability graphs and reliability graphs is apparent from the presence of repair links in availability graphs. Subsequently, a reliability analysis becomes an availability analysis when repair links, with repair rate parameters, are included in the analysis.

Claim Rejections - 35 USC § 101

35 U.S.C. § 101 reads as follows:

> Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

2. Claims 1-7 are rejected under 35 U.S.C. § 101 because the claimed invention is directed to non-statutory subject matter. MPEP 2106 reads as follows:

> Apart from the utility requirement of 35 U.S.C. 101, usefulness under the patent eligibility standard requires significant functionality to be present to satisfy the useful result aspect of the practical application requirement. See Arrhythmia, 958 F.2d at 1057, 22 USPQ2d at 1036. Merely claiming nonfunctional descriptive material stored in a computer-readable medium does not make the invention eligible for patenting. For example, a claim directed to a word processing file stored on a disk may satisfy the utility requirement of 35 U.S.C. 101 since the information stored may have some "real world" value. However, the mere fact that the claim may satisfy the utility requirement of 35 U.S.C. 101 does not mean that a useful result is achieved under the practical application requirement. The claimed invention as a whole must produce a "useful, concrete and tangible" result to have a practical application.

Claims 1-7 are directed toward "an availability model" and "a network model", both of which are data abstractions. These claims are drafted in the form of an apparatus or manufacture claim, however the claimed invention is an intangible arrangement of data. As such, the claimed invention is nonfunctional descriptive material and therefore nonstatutory.

3. Claims 8-16 are rejected under 35 U.S.C. § 101 because the claimed invention is directed to non-statutory subject matter. MPEP 2106 (IV)(B)(2)(b) reads as follows:

> To be statutory, a claimed computer-related process must either: (A) result in a physical transformation outside the computer for which a practical application in the technological arts is either disclosed in the specification or would have been known to a skilled artisan (discussed in i) below), or (B) be limited to a practical application within the technological arts (discussed in ii) below). See *Diamond v. Diehr*, 450 U.S. at 183-84, 209 USPQ at 6 (quoting *Cochrane v. Deener*, 94 U.S. 780, 787-88 (1877))

MPEP 2106 (IV)(B)(2)(b)(ii) reads as follows:

> For such subject matter to be statutory, the claimed process must be limited to a practical application of the abstract idea or mathematical algorithm in the technological arts. See *Alappat*, 33 F.3d at 1543, 31 USPQ2d at 1556-57 (quoting *Diamond v. Diehr*, 450 U.S. at 192, 209 USPQ at 10). See also *Alappat* 33 F.3d at 1569, 31 USPQ2d at 1578-79 (Newman, J., concurring) ("unpatentability of the principle does not defeat patentability of its practical applications") (citing *O'Reilly v. Morse*, 56 U.S. (15 How.) at 114-19). A claim is limited to a practical application when the method, as claimed, produces a concrete, tangible and useful result; i.e., the method recites a step or act of producing something that is concrete, tangible and useful. See AT &T, 172 F.3d at 1358, 50 USPQ2d at 1452.

Claims 8-16 recite various methods that do not produce a useful, concrete, and tangible result. They do recite methods that manipulate abstractions, such as a model of a node in a network or modeling an error in a network, however they are not limited to the technological arts. For example, granting the broadest reasonable interpretation to the claims, a person with a pencil and paper could infringe the recited methods. The Examiner respectfully suggests claiming these as computer-implemented methods, thus limiting them to the technological arts.

Please see MPEP 2111 regarding granting claims their broadest reasonable interpretation.

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

4. Claims 1-2 are rejected under 35 U.S.C. § 103(a) as being unpatentable over US Patent No. 5,014,220 to McMann et al. (McMann) in view of "Understanding Fault-Tolerant Distributed Systems" by Flaviu Cristian (Cristian) (cited on PTO-892 paper number 20040823).

McMann teaches a system and method for generating a reliability model for a complex system having different classes of failures ["*The present invention provides a reliability model for use by a reliability analysis tool.*" (column 5, lines 37-54); "*A system in SURE is defined as a state space description: the set of all feasible states of the system, given an initial state. State transitions, in SURE, describe the <u>occurrence of faults and fault recovery actions</u> that cause the system to change from one state to another,*" (column 6, lines 3-8, emphasis added); "*For highly reliable system, additional functions are incorporated into the architecture for <u>failure detection, isolation, and recovery (FDIR)</u>,*" (column 5, lines 54-56, emphasis added)];

The reliability model includes a model for defining expected failure rates and time to recover from the expected failures for components of the platform ["*Given the state space description, including an identification of the initial state and those states that represent an unreliable system, SURE computes the <u>upper and lower bounds on system reliability</u> and*

provides an enumeration of all system failures," (column 6, lines 8-12, emphasis added); transitions include recovery actions, as in "*The failure modes and FDIR attributes are described to ASSIST as transitions in the form of logical statements,*" (column 6, lines 20-21, emphasis added); failures and recoveries, represented by transitions, are time and rate dependent, as in "*Another critical aspect of FMEA is concerned with the effects of multiple failures on the system and the effects of nearly simultaneous failures – a particular state of vulnerability in which a second failure may occur before the system can recover from the first failure. These time dependencies contribute to the difficulty of an accurate reliability analysis",* (column 5, lines 58-64)];

A reliability model within the platform reliability model, including an aggregated failure rate for each class of failures and an aggregated repair time for each class of failures for at least one component ["*To manage the analysis complexity, a system may be divided into sets of components.* [...] *This component then becomes a lowest level component in a new aggregate model that also accounts for dependencies among the sets,*" (column 7, lines 20-30, emphasis added)].

McMann does not expressly teach the specific failure modes recited in the claim, however McMann does explicitly teach that the disclosed invention is a framework for developing "a model for a system of virtually any complexity" (column 2, lines 11-14). McMann further teaches the suitability of the reliability modeling system and method to other disciplines ["*These units may correspond to a physical hardware device or may refer to assemblies of units for which composite failure modes are identified. The units have been referred to in literature by various nomenclature including systems and subsystems, assemblies*

and subassemblies, components and subcomponents, structures and substructures, etc." (column 6, lines 56-63)]. McMann also teaches a computer hardware example (column 7, lines 4-19)].

Cristian teaches the four failure modes recited in the claim as known in the art:

"failures that can be corrected internally with no loss of service" [*"A* timing *failure occurs when the server's response is functionally correct but untimely – the response occurs outside the real-time interval specified,"* (page 58, center and right columns)],

"failures that can be corrected by a restart with no loss of state" [*"If, after a first omission to produce output, <u>a server omits to produce output to subsequent inputs until its restart</u>, the server is said to suffer a* crash *failure*"; and *"A pause-crash occurs when a server restarts in the state it had before the crash,"* (page 58, right column, emphasis added)],

"failures that can be corrected by a restart with loss of state" [*"If, after a first omission to produce output, <u>a server omits to produce output to subsequent inputs until its restart</u>, the server is said to suffer a* crash *failure*"; and *"An amnesia-crash occurs when the server restarts in a predefined initial state that does not depend on the inputs seen before the crash,"* (page 58, right column, emphasis added)],

"failures that can be corrected by fail over" [*"A halting-crash occurs when a crashed server never restarts,"* (page 58, right column, emphasis added); Official notice is taken that a person of ordinary skill in the art would recognize that a "fail over" is the necessary recovery action from a "halting-crash"].

It would have been obvious to a person of ordinary skill in the art at the time of Applicants' invention to combine the failure modes taught by Cristian for the purposes of modeling the availability of software and hardware systems with the reliability model generation

system of McMann to arrive at the claimed invention; an availability model for software with the particular failure modes. Although much of McMann is directed toward a reliability model, McMann also provides sufficient teaching of recovery actions to suggest adapting the invention into an availability model, such an adaptation being described as known in the art by Malek. Motivation to combine the failure modes taught by Cristian with the reliability model generation of McMann would be found in the nature of the problem to be solved ["*The task of designing and understanding fault-tolerant distributed system architectures is notoriously difficult*", (Cristian, page 57, left column); which complements "*The use of digital systems and redundancy management schemes to satisfy flight control system requirements of high performance aircraft has increased both the number of implementation alternatives and the overall system design complexity. Consequently, a comprehensive reliability analysis of each candidate architecture becomes tedious, time-consuming, and costly*", (McMann, column 1, lines 29-35)], that is, designing and understanding a complex system.

Regarding claim 2, McMann teaches platform parameters define platform problems causing failures ["*Failure modes of subcomponents are combined according to their severity and common effects on a higher level component. <u>These failure modes are used to define a model of the component at the higher level.</u> This component then becomes a lowest level component in a new aggregate model that also accounts for dependencies among the sets*", (column 7, lines 24-30, emphasis added)] and affecting recovery times related to the platform problems ["*This component then becomes a lowest level component in <u>a new aggregate model that also accounts for dependencies among the sets</u>*", (column 7, lines 28-30, emphasis added); "*Therefore, to*

determine the sequence of component failures that contribute to a particular undesirable condition, a Failure Mode Effect Analysis (FMEA) is performed that traces the effects of component failures according to component interactions. For highly reliable systems, additional functions are incorporated into the architecture for failure detection, isolation, and recovery (FDIR)", (column 5, lines 49-56, emphasis added)] and wherein at least a portion of the platform parameters are used to determine the aggregated repair time ["*Once a top level reliability model 210 is defined, further reduction techniques are applied by the model reducer/encoder 204 of FIG. 2, to reduce the model state space and encode the global model into the ASSIST syntax from which the SURE model is built"*, (column 9, lines 18-23, emphasis added); "*A system in SURE is defined as a state space description: the set of all feasible states of the system, given an initial state. State transitions, in SURE, describe the occurrence of faults and fault recovery actions that cause the system to change from one state to another"*, (column 6, lines 3-8, emphasis added)].

5. Claim 3 is rejected under 35 U.S.C. § 103(a) as being unpatentable over McMann in view of Cristian as applied to claim 1 above, and further in view of US Patent No. 4,870,474 to Rutenberg.

McMann teaches the capability to include a hardware component availability model within the platform availability model ["*To manage the analysis complexity, a system may be divided into sets of components*", (column 7, lines 20-21)] and provides a multiprocessor

example (column 7, lines 4-19). While McMann teaches the capability to include a hardware component availability model, such a model is not explicitly disclosed.

Cristian teaches various faults, including hardware faults, but does not explicitly disclose combining a hardware component availability model within a platform availability model.

Rutenberg teaches a fault-tree analysis which can detect all latent hardware and software design defects that could cause unanticipated critical failure of a complex software controlled electronic system (abstract). Rutenberg explicitly teaches motivation for performing a combined hardware and software fault analysis ["*As discussed above, a complete analysis of the critical failure potential of a design can only result from an understanding of all the possible interactions between the system hardware and its control software*", (column 6, lines 7-11)].

In light of the Rutenberg teachings and motivation, it would have been obvious to a person of ordinary skill in the art at the time of Applicants' invention to include a hardware component availability model when using the system and method of generating a reliability model taught by McMann. Such a hardware component availability model would be yet another set of components of the larger system, as taught by McMann. The motivation for making the combination could be as explicitly taught by Rutenberg, that is, to perform a complete analysis of the interactions between the hardware and software of a system.

6. Claim 4 is rejected under 35 U.S.C. § 103(a) as being unpatentable over McMann in view of Cristian as applied to claim 1 above, and further in view of "Survey of Software Tools for

Evaluating Reliability, Availability, and Serviceability" by Allen M. Johnson, Jr., and Mirslaw Malek (Malek).

McMann does not explicitly teach that the aggregated repair time includes a time to detect and identify an error associated with running the at least one software component on said platform.

Cristian does not explicitly teach that time to detect and identify an error contributes to an aggregated repair time.

Malek teaches numerous concepts known in the art related to availability and reliability modeling of computing systems and software (section 1.2, "Service Cost and Repair Time Model"), in particular a model for mean time to repair (MTTR) (page 223, left column). This model includes "a time to detect and identify an error", broken into several components such as $T_A(i)$, "average time to talk to customer and obtain the fault symptoms, identification of failing unit, and any other preparation required in the *i*th month"; T_B, "time required to run diagnostics or analyze information logged at the time of the error to determine the fault symptoms"; U_{diag}, "application factor to obtain the additional time required when the diagnostics or logout analysis are not effective in isolating the problem to a single RU" (*replaceable units*, page 232, right column); and P_{isol}, "probability that the error symptoms uniquely identify the failing RU (depending upon the maintenance strategy applied)". The goal of Malek's MTTR model is to accurately calculate the hours spent servicing a given type of system (page 232, right column).

It would have been obvious to a person of ordinary skill in the art to combine the accurate MTTR model taught by Malek with the reliability model generation system and method taught by McMann to achieve a more accurate model of the recovery actions in the system. Such a

combination could be achieved by incorporating the MTTR model calculations taught by Malek into the state transitions of the SURE model generated by McMann's system and method. Motivation to combine would be apparent to a person of ordinary skill in the art as a result of the accuracy and comprehensiveness of Malek's MTTR model.

7. Claim 5 is rejected under 35 U.S.C. § 103(a) as being unpatentable over McMann in view of Cristian as applied to claim 1 above, and further in view of "Availability analysis of a certain class of distributed computer systems under the influence of hardware and software faults" by G.D. Hassapis (Hassapis)

McMann teaches the capability to generate a reliability model where the system is a node in a network [*"Briefly described, the present invention contemplates a reliability model generator which automatically generates a composite reliability model for a system of virtually any complexity"*, (column 2, lines 11-14); *"To manage the analysis complexity, a system may be divided into sets of components*", (column 7, lines 20-21)]. While McMann teaches the capability to generate a reliability model for node in a network, such a model is not explicitly disclosed.

Cristian teaches various faults, including server faults, but does not explicitly disclose a reliability model including a node in a network.

Hassapis teaches an availability model for a computer platform with at least one software component [*"...assess the availability when the system is subjected to the combined effects of hardware and software faults either during its normal operating time or repair time."* (abstract)]

wherein the hardware platform is a node in a network ["*This theory has been made more appropriate for the type of software used in the distributed process control systems and has been extended by incorporating the state of the computer hardware at time* t *explicitly*", (page 524, right column, emphasis added); *network processor, network interface,* etc., page 527, Fig. 1].

It would have been obvious to a person of ordinary skill in the art at the time of Applicants' invention to combine the teachings of Hassapis, regarding a combined hardware software availability analysis wherein the hardware is a node in a network, with the reliability model generation system and method of McMann in order to accurately assess the availability of a complex system such as a distributed computing system. The combination could be achieved representing a node in a network as the system of components described by McMann. Motivation to do so would be found in the nature of the problem to be solved, such as the need to analyze the availability of a node in a network, wherein the node comprises both hardware and software.

8. Claim 6 is rejected under 35 U.S.C. § 103(a) as being unpatentable over McMann in view of Hassapis.

McMann teaches a system and method for generating a reliability model for a complex system having different classes of failures ["*The present invention provides a reliability model for use by a reliability analysis tool.*" (column 5, lines 37-54); "*A system in SURE is defined as a state space description: the set of all feasible states of the system, given an initial state. State transitions, in SURE, describe the occurrence of faults and fault recovery actions that cause the*

system to change from one state to another," (column 6, lines 3-8, emphasis added); "*For highly reliable system, additional functions are incorporated into the architecture for failure detection, isolation, and recovery (FDIR),*" (column 5, lines 54-56, emphasis added)];

The reliability model includes a model for defining expected failure rates and time to recover from the expected failures for components of the platform ["*Given the state space description, including an identification of the initial state and those states that represent an unreliable system, SURE computes the upper and lower bounds on system reliability and provides an enumeration of all system failures,*" (column 6, lines 8-12, emphasis added); transitions include recovery actions, as in "*The failure modes and FDIR attributes are described to ASSIST as transitions in the form of logical statements,*" (column 6, lines 20-21, emphasis added); failures and recoveries, represented by transitions, are time and rate dependent, as in "*Another critical aspect of FMEA is concerned with the effects of multiple failures on the system and the effects of nearly simultaneous failures – a particular state of vulnerability in which a second failure may occur before the system can recover from the first failure. These time dependencies contribute to the difficulty of an accurate reliability analysis*", (column 5, lines 58-64)].

Official Notice is taken that in the case of a network or distributed computing system, it is known in the art that node reboot time significantly contributes to the recovery time of the system and should therefore be contemplated as parameters of the corresponding recovery actions.

A reliability model within the platform reliability model, including an aggregated failure rate for each class of failures and an aggregated repair time for each class of failures for at least

one component ["*To manage the analysis complexity, a system may be divided into sets of components.* [...] *<u>This component then becomes a lowest level component in a new aggregate model</u> that also accounts for dependencies among the sets,*" (column 7, lines 20-30, emphasis added)].

McMann teaches the capability to generate a reliability model where the system is a node in a network ["*Briefly described, the present invention contemplates a reliability model generator which automatically generates a composite reliability model for a system of virtually any complexity*", (column 2, lines 11-14); "*To manage the analysis complexity, a system may be divided into sets of components*", (column 7, lines 20-21)]. While McMann teaches the capability to generate a reliability model for node in a network, such a model is not explicitly disclosed.

Hassapis teaches an availability model for a computer platform with at least one software component [*"...assess the availability when the system is subjected to the combined effects of hardware and software faults either during its normal operating time or repair time."* (abstract)] wherein the hardware platform is a node in a network ["*This theory has been made more appropriate for the type of <u>software used in the distributed process control systems</u> and has been extended by incorporating the state of the computer hardware at time* t *explicitly*", (page 524, right column, emphasis added); *network processor, network interface, etc.*, page 527, Fig. 1].

It would have been obvious to a person of ordinary skill in the art at the time of Applicants' invention to combine the teachings of Hassapis, regarding a combined hardware software availability analysis wherein the hardware is a node in a network, with the reliability model generation system and method of McMann in order to accurately assess the availability of

a complex system such as a distributed computing system. The combination could be achieved representing a node in a network as the system of components described by McMann. Motivation to do so would be found in the nature of the problem to be solved, such as the need to analyze the availability of a node in a network, wherein the node comprises both hardware and software.

Regarding claim 7, McMann teaches the capability to include a hardware component availability model within the platform availability model ["*To manage the analysis complexity, a system may be divided into sets of components*", (column 7, lines 20-21)] and provides a multiprocessor example (column 7, lines 4-19). The combination formed in the rejection of claim 6 involves representing a node in a network as a system or set of components in the reliability model generation system and method of McMann. As a node in a network is a hardware component, that combination also teaches the limitations of claim 7.

9. Claims 8 and 18 are rejected under 35 U.S.C. § 103(a) as being unpatentable over McMann in view of Cristian.

McMann teaches a system and method for generating a reliability model for a complex system having different classes of failures ["*The present invention provides a reliability model for use by a reliability analysis tool.*" (column 5, lines 37-54); "*A system in SURE is defined as a state space description: the set of all feasible states of the system, given an initial state. State transitions, in SURE, describe the occurrence of faults and fault recovery actions that cause the*

system to change from one state to another," (column 6, lines 3-8, emphasis added); "*For highly reliable system, additional functions are incorporated into the architecture for failure detection, isolation, and recovery (FDIR),*" (column 5, lines 54-56, emphasis added)];

The reliability model includes a model for defining expected failure rates and time to recover from the expected failures for components of the platform ["*Given the state space description, including an identification of the initial state and those states that represent an unreliable system, SURE computes the upper and lower bounds on system reliability and provides an enumeration of all system failures,*" (column 6, lines 8-12, emphasis added); transitions include recovery actions, as in "*The failure modes and FDIR attributes are described to ASSIST as transitions in the form of logical statements,*" (column 6, lines 20-21, emphasis added); failures and recoveries, represented by transitions, are time and rate dependent, as in "*Another critical aspect of FMEA is concerned with the effects of multiple failures on the system and the effects of nearly simultaneous failures – a particular state of vulnerability in which a second failure may occur before the system can recover from the first failure. These time dependencies contribute to the difficulty of an accurate reliability analysis*", (column 5, lines 58-64)];

Failure rates and recovery rates are used to generate state transition parameters ("*State transitions, in SURE, describe the occurrence of faults and fault recovery actions that cause the system to change from one state to another*", (column 6, lines 5-8)];

A reliability model within the platform reliability model, including an aggregated failure rate for each class of failures and an aggregated repair time for each class of failures for at least one component ["*To manage the analysis complexity, a system may be divided into sets of*

components. [...] *<u>This component then becomes a lowest level component in a new aggregate model</u> that also accounts for dependencies among the sets,*" (column 7, lines 20-30, emphasis added)].

McMann does not expressly teach the specific errors recited in the claim, however McMann does explicitly teach that the disclosed invention is a framework for developing "a model for a system of virtually any complexity" (column 2, lines 11-14). McMann further teaches the suitability of the reliability modeling system and method to other disciplines ["*These units may correspond to a physical hardware device or may refer to assemblies of units for which composite failure modes are identified. The units have been referred to in literature by various nomenclature including systems and subsystems, assemblies and subassemblies, components and subcomponents, structures and substructures, etc.*" (column 6, lines 56-63)].

Cristian teaches the errors recited in the claim as known in the art:

"warm recoverable errors" comprise application failures that can be corrected by a restart without loss of state of the application ["*If, after a first omission to produce output, <u>a server omits to produce output to subsequent inputs until its restart</u>, the server is said to suffer a* crash *failure*"; and "*A pause-crash occurs when <u>a server restarts in the state it had before the crash</u>,*" (page 58, right column, emphasis added)], and

"non-warm recoverable errors" comprise application failures that can be corrected by a restart with loss of state in the application ["*If, after a first omission to produce output, <u>a server omits to produce output to subsequent inputs until its restart</u>, the server is said to suffer a* crash *failure*"; and "*An amnesia-crash occurs when the <u>server restarts in a predefined initial state that does not depend on the inputs seen before the crash</u>,*" (page 58, right column, emphasis added)].

It would have been obvious to a person of ordinary skill in the art at the time of Applicants' invention to combine the errors taught by Cristian with the reliability model generation system and method of McMann in order to better analyze and understand a complex system, such as a distributed computing system contemplated by Cristian. Such a system would comprise a model of the network defining the distributed computing system. The combination could be achieved by modeling the distributed system using the method taught by McMann, where the various subcomponents correspond to individual computer systems and the hardware and software on those systems.

10. Claim 18 recites a computer program product comprising computer readable code for performing the method of claim 8. As McMann is a computer-implemented method (Fig. 2), claim 18 is rejected for the same reasons and with the same combination formed in the rejection of claim 8.

11. Claims 9-12 are rejected under 35 U.S.C. § 103(a) as being unpatentable over McMann in view of Cristian, and further in view of Malek.

Neither Cristian nor McMann explicitly teach determining a fraction of recovery failures for warm or non-warm recoverable errors as recited in the claim. However, as noted in the rejection of claim 8, Cristian does teach "warm recoverable errors" and "non-warm recoverable errors".

Malek teaches contributing factors to the mean time to repair (MTTR) which are functionally equivalent to a fraction of recovery failures, such as T_B, "time required to run

diagnostics or analyze information logged at the time of the error to determine the fault symptoms"; P_{diag}, "probability that the diagnostics or logout analysis will be effective in determining the fault symptoms"; and T_E, "time required to run the diagnostics to verify that the problem has been fixed" (page 233, left and right columns). Malek teaches MTTR from the perspective that the fault will be eventually corrected; as such, a "recovery failure" is represented by the inverse of P_{diag}, that a fault will be incorrectly diagnosed and time treating it, $T_D(i)$ and T_E, will be lost. Malek considers the probability of misdiagnosis of the fault (a probability being a number between 0 and 1), which leads directly to a failure to recover.

It would have been obvious to a person of ordinary skill in the art to combine the MTTR model taught by Malek with the combined reliability model generation system and method of McMann in view of Cristian in order to more accurately model the state transitions from failure to operational, especially when concerned with simultaneous failures (McMann, column 5, lines 58-63). The combination could be achieved by using Malek's MTTR model when computing the state transitions for a recovery action.

Regarding claims 13-15, McMann teaches that the reliability model includes a model for defining parameters of the node, such as expected failure rates and time to recover from the expected failures for components of the platform ["*Given the state space description, including an identification of the initial state and those states that represent an unreliable system, SURE computes the* <u>*upper and lower bounds on system reliability*</u> *and provides an enumeration of all system failures,*" (column 6, lines 8-12, emphasis added); transitions include recovery actions, as

in "*The failure modes and <u>FDIR attributes</u> are described to ASSIST as <u>transitions in the form of logical statements</u>,*" (column 6, lines 20-21, emphasis added); failures and recoveries, represented by transitions, are time and rate dependent, as in "*Another critical aspect of FMEA is concerned with the effects of multiple failures on the system and the effects of nearly simultaneous failures – a particular state of vulnerability in which a second failure may occur before the system can recover from the first failure. These time dependencies contribute to the difficulty of an accurate reliability analysis*", (column 5, lines 58-64)]. Failure rates and recovery rates are used to generate state transition parameters ("*State transitions, in SURE, describe the occurrence of faults and fault recovery actions that cause the system to change from one state to another*", (column 6, lines 5-8)]. McMann explicitly teaches considering the recovery times for subcomponents and components.

Official Notice is taken that in the case of a network or distributed computing system, it is known in the art that subcomponent (node) reboot time and component (network) reboot time significantly contributes to the recovery time of the system and should therefore be contemplated as parameters of the corresponding recovery actions.

12. Claims 16 and 19 are rejected under 35 U.S.C. § 103(a) as being unpatentable over McMann in view of Malek.

McMann teaches a system and method for generating a reliability model for a complex system having different classes of failures ["*The present invention provides a reliability model for use by a reliability analysis tool.*" (column 5, lines 37-54); "*A system in SURE is defined as a*

state space description: the set of all feasible states of the system, given an initial state. State transitions, in SURE, describe the *__occurrence of faults and fault recovery actions__* *that cause the system to change from one state to another,*" (column 6, lines 3-8, emphasis added); "*For highly reliable system, additional functions are incorporated into the architecture for* *__failure detection, isolation, and recovery__* *(FDIR),*" (column 5, lines 54-56, emphasis added)];

The reliability model defines a recoverable state for a modeled error ["*State transitions, in SURE, describe the occurrence of faults and* *__fault recovery actions that cause the system to change from one state to another__*. *Given the state space description, including an identification of the initial state and those* *__states that represent an unreliable system__*, *SURE computes the upper and lower bounds on system reliability and provides an enumeration of all system failures,*" (column 6, lines 5-12, emphasis added)]; and

The reliability model determines a failure rate for said error and a recovery rate for said error [See above, also failures and recoveries, represented by transitions, are time and rate dependent, as in "*Another critical aspect of FMEA is concerned with the effects of multiple failures on the system and the effects of nearly simultaneous failures – a particular state of vulnerability in which a second failure may occur before the system can recover from the first failure. These time dependencies contribute to the difficulty of an accurate reliability analysis*", (column 5, lines 58-64)].

McMann does not explicitly teach determining a fraction of recovery failures for warm or non-warm recoverable errors as recited in the claim.

Malek teaches contributing factors to the mean time to repair (MTTR) which are functionally equivalent to a fraction of recovery failures, such as T_B, "time required to run

diagnostics or analyze information logged at the time of the error to determine the fault symptoms"; P_{diag}, "probability that the diagnostics or logout analysis will be effective in determining the fault symptoms"; and T_E, "time required to run the diagnostics to verify that the problem has been fixed" (page 233, left and right columns). Malek teaches MTTR from the perspective that the fault will be eventually corrected; as such, a "recovery failure" is represented by the inverse of P_{diag}, that a fault will be incorrectly diagnosed and time treating it, $T_D(i)$ and T_E, will be lost. Malek considers the probability of misdiagnosis of the fault (a probability being a number between 0 and 1), which leads directly to a failure to recover.

It would have been obvious to a person of ordinary skill in the art to combine the MTTR model taught by Malek with the combined reliability model generation system and method of McMann in view of Cristian in order to more accurately model the state transitions from failure to operational, especially when concerned with simultaneous failures (McMann, column 5, lines 58-64). The combination could be achieved by using Malek's MTTR model when computing the state transitions for a recovery action.

13. Claim 19 recites a computer program product comprising computer readable code for performing the method of claim 16. As McMann is a computer-implemented method (Fig. 2), claim 19 is rejected for the same reasons and with the same combination formed in the rejection of claim 16.

Conclusion

Art considered pertinent by the examiner but not applied has been cited on form PTO-892.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Jason Proctor whose telephone number is (571) 272-3713. The examiner can normally be reached on 8:30 am-4:30 pm M-F.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Leo Picard can be reached at (571) 272-3749. The fax phone number for the organization where this application or proceeding is assigned is (571) 273-3713.

Any inquiry of a general nature or relating to the status of this application should be directed to the TC 2100 Group receptionist: 571-272-2100. Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Jason Proctor
Examiner
Art Unit 2123

jsp

TC 2100
AU 2123
Primary Examiner